AMENDMENTS TO THE CLAIMS

Claims 1-40 were pending at the time of the Office Action.

Claims 1, 8, 14, 18, 19, 23, 31, 38, and 40 are amended.

Claims 1-40 remain pending.

1. (Currently amended) One or more computer readable storage media having stored thereon a plurality of instructions that, when executed by one or more processors, causes the one or more processors to perform a method ~~determine whether an input value matches any of a plurality of target values by performing acts~~ including:

generating a hash key based on an ~~the~~ input value;

separating the hash key into a plurality of portions;

indexing into each of a plurality of sub-hashes using one of the plurality of portions, each location in each of the plurality of sub-hashes containing a multiple-bit value, each bit corresponding to one of the plurality of target values;

identifying a plurality of values from the plurality of sub-hashes based on the indexing;

combining the plurality of values to generate a hash result, wherein each bit in the hash result corresponds to one of the plurality of target values; ~~and~~

for each bit in the hash result that is set, comparing the input value to the corresponding target value; and

allowing access based on ~~providing an indication~~ whether the values match.

2. (Original) One or more computer readable media as recited in claim 1, wherein the number of target values in the plurality of target values is equal to the number of bits in the hash result.

3. (Original) One or more computer readable media as recited in claim 1, wherein a maximum number of target values in the plurality of target values is equal to the number of bits in the result value.

4. (Original) One or more computer readable media as recited in claim 1, wherein a maximum number of target values in the plurality of target values is equal to the number of bits in each of a plurality of locations of the plurality of sub-hashes that can be indexed.

5. (Original) One or more computer readable media as recited in claim 1, wherein the separating comprises separating the hash key into two portions.

6. (Original) One or more computer readable media as recited in claim 1, wherein the separating comprises separating the hash key into a plurality of contiguous and equal portions.

7. (Original) One or more computer readable media as recited in claim 1, wherein the combining comprises performing a bitwise logical ANDing of the plurality of values.

8. (Currently Amended) A hashing architecture implemented in hardware for determining whether to allow access to an access controlled object ~~an input value matches any of a plurality of target values~~, comprising:

a plurality of sub-hashes;

a plurality of sub-hash indexes, each index being generated from a hash key and used to index into one of the plurality of sub-hashes, each location in each of the plurality of sub-hashes containing a multiple-bit value, each bit corresponding to one of the plurality of target values; and

a combiner coupled to receive values from the plurality of sub-hashes based on the plurality of sub-hash indexes, and to generate a hash result by combining the received values; and

a comparator coupled to receive the hash result and to

determine which of the plurality of target values to compare to the input value;

compare the input value to at least one of the plurality of target values; and

allowing access based on the comparison.

9. (Original) A hashing architecture as recited in claim 8, wherein the combiner comprises a combinatorial logic component to perform a bitwise logical ANDing of the values received from the plurality of sub-hashes.

10. (Original) A hashing architecture as recited in claim 8, wherein the hashing architecture is implemented in software.

11. (Original) A hashing architecture as recited in claim 8, wherein the hashing architecture is implemented in firmware.

12. (Previously Presented) A hashing architecture as recited in claim 8, wherein the hashing architecture is implemented in a computing device.

13. (Original) A hashing architecture as recited in claim 8, wherein the plurality of sub-hash indexes are generated by separating the hash key into a plurality of equal portions.

14. (Currently Amended) A method ~~of determining whether an input value matches any of a plurality of target values~~, comprising:

generating a plurality of sub-hash keys based on a hash key, the hash key being based on the input value;

identifying a plurality of values from a plurality of sub-hashes by indexing into each of the plurality of sub-hashes using one of the plurality of sub-hash keys, each location in each of the plurality of sub-hashes containing a multiple-bit value, each bit corresponding to one of the plurality of target values; ~~and~~

generating a hash result based on the plurality of values; ~~and~~

determining based on the hash result which of the plurality of target values to compare to the input value;

comparing the input value to at least one of the plurality of target values; and

allowing access based on the comparison.

15. (Original) A method as recited in claim 14, further comprising generating the hash key prior to generating the plurality of sub-hash keys.

16. (Original) A method as recited in claim 14, wherein the generating the plurality of sub-hash keys comprises separating the hash key into a plurality of equal portions.

17. (Original) A method as recited in claim 14, wherein the generating the hash result comprises performing a bit-by-bit logical ANDing of the plurality of values.

18. (Currently Amended) One or more computer readable media storing ~~including~~ a computer program that is executable by a processor to perform the method recited in claim 14.

19. (Currently Amended) One or more computer readable storage media having stored thereon a plurality of instructions that, when executed by one or more processors, ~~determine whether a security identifier of an access control element matches any of a plurality of security identifiers of a security token by causing~~ cause the one or more processors to perform a method ~~acts~~ including:

generating a hash key based on an ~~the~~ access control element security identifier;

separating the hash key into a first portion and a second portion;

indexing into a first sub-hash using the first portion to identify a first sub-hash value;

indexing into a second sub-hash using the second portion to identify a second sub-hash value;

combining the first sub-hash value and the second sub-hash value to generate a result value, wherein each bit in the result value corresponds to one of a ~~the~~ plurality of security token security identifiers; and

for each bit in the result value that is set,

comparing the access control element security identifier to the corresponding security token security identifier; and ~~providing an indication whether the values match~~

allowing access based on whether the access control security identifier matches the corresponding security token identifier.

20. (Original) One or more computer readable media as recited in claim 19, wherein the generating comprises generating the hash key by selecting a portion of the access control element security identifier.

21. (Original) One or more computer readable media as recited in claim 19, wherein the separating comprises separating the hash key into two portions that include an equal number of bits and that are contiguous.

22. (Original) One or more computer readable media as recited in claim 19, wherein the combining comprises bitwise ANDing together the first sub-hash value and the second sub-hash value.

23. (Currently Amended) A method ~~of determining whether an input security identifier matches one or more of a plurality of target security identifiers, the method~~ comprising:

generating a plurality of sub-hash indexes based on a hash key, the hash key being based on an ~~the~~ input security identifier;

indexing into each of a plurality of sub-hashes using a respective one of a ~~the~~ plurality of sub-hash indexes, each location in each of the plurality of sub-hashes containing a multiple-bit value, each bit corresponding to one of the plurality of target security identifiers;

generating a result hash value by combining the plurality of values resulting from indexing into the plurality of sub-hashes, wherein each of the plurality of target security identifiers corresponds to a portion of the result hash value; ~~and~~

comparing the input security identifier to at least one of the plurality of target security identifiers that corresponds to a portion of the result hash value having a particular value ~~to determine whether a match exists~~; and

allowing access based on whether the input security identifier matches one or more of the plurality of target security identifiers.

24. (Original) A method as recited in claim 23, wherein the particular value comprises a value of one.

25. (Original) A method as recited in claim 23, wherein each portion is a bit of the result hash value.

26. (Original) A method as recited in claim 23, wherein the generating a plurality of sub-hash indexes comprises:

selecting a portion of the input security identifier;

separating the portion into two equal and contiguous sub-portions; and

using each of the sub-portions as one of the plurality of sub-hash indexes.

27. (Original) A method as recited in claim 23, wherein the generating the result hash value comprises generating the result hash value by performing a bitwise logical ANDing of the plurality of values.

28. (Original) A method as recited in claim 23, wherein the input security identifier comprises an access control security identifier and each of the plurality of target security identifiers comprises a security token security identifier.

29. (Original) A method as recited in claim 23, wherein the input security identifier comprises a security token security identifier and each of the plurality of target security identifiers comprises an access control security identifier.

30. (Original) One or more computer readable media including a computer program that is executable by a processor to perform the method recited in claim 23.

31. (Currently Amended) A computer-based system comprising:

a plurality of security token security identifiers corresponding to a user;

a plurality of access control security identifiers corresponding to an object;

a plurality of sub-hashes, each location in each of the plurality of sub-hashes containing a multiple-bit value, each bit corresponding to one of the plurality of target security identifiers; and

an access controller to determine whether any of the plurality of security token security identifiers match any of the plurality of access control security identifiers by, for each of the plurality of access control security identifiers,

generating a plurality of sub-hash indexes based on a hash key,

indexing into each of the plurality of sub-hashes using a respective one of the plurality of sub-hash indexes,

identifying a plurality of values from the plurality of sub-hashes based on the indexing,

combining the plurality of values to generate a hash result value, wherein each bit in the hash result value corresponds to one of the plurality of security token security identifiers, and

for each bit in the result value that is set,

comparing the access control security identifier to the corresponding security token security identifier to determine whether the values match; and

allowing the user access to the object based on whether the values match.

32. (Original) A system as recited in claim 31, wherein the plurality of sub-hashes comprise two sub-hashes.

33. (Original) A system as recited in claim 31, wherein each bit in the result value that is set has a value of one.

34. (Original) A system as recited in claim 31, wherein the combining comprises bitwise logically ANDing together the plurality of values.

35. (Original) A system as recited in claim 31, wherein each of the plurality of values and the hash result value each includes a number of bits equal to a maximum number of security token security identifiers that can be included in the plurality of security token security identifiers.

36. (Original) A system as recited in claim 31, wherein the system comprises an operating system.

37. (Original) A system as recited in claim 31, wherein the system comprises a resource manager that is not a part of an operating system.

38. (Currently Amended) A method ~~for determining whether an input value matches a target value~~, comprising:

for each sub-hash in a plurality of sub-hashes that can be used together to generate a hash result for determining whether an ~~the~~ input value matches a ~~the~~ target value, wherein each location in each of the plurality of sub-hashes contains a multiple-bit value, each bit corresponding to one of a ~~the~~ plurality of target values,

(a) identifying a bit in a location of the sub-hash,

(b) identifying, in a source value, a plurality of bits corresponding to the sub-hash,

(c) comparing an identifier of the location to the plurality of bits,

(d) setting the bit if the identifier of the location matches the plurality of bits, and otherwise clearing the bit, and

(e) repeating acts (a), (b), (c), and (d) for each of a plurality of bits in the location of the sub-hash; and

allowing access based on whether the input value matches the target value.

39. (Original) A method as recited in claim 38, wherein the plurality of bits correspond to part of a portion of the source value that will be used to generate a hash value.

40. (Currently Amended) One or more computer readable media storing ~~including~~ a computer program that is executable by a processor to perform the method recited in claim 38.